CSE-1006| Lab Assignment-4 | L29-L30

Academic year: *2021-2022* | Semester: *WIN*

Faculty Name: ***Dr Arunkumar Gopu*** | Date: ***18/5/2022***

Student name: ***M.Taran*** | Reg. no.: ***19BCE7346***

importing

In [1]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [5]:

```
data = pd.read_csv("/content/Dataset.csv")
data
```

Out[5]:

	Gender	Age	Education Level	Institution Type	IT Student	Location	Load to shedding	Financial Condition	Internet Type	N
0	Boy	21 to 25	University	Non Government	No	Yes	Low	Mid	Wifi	
1	Girl	21 to 25	University	Non Government	No	Yes	High	Mid	Mobile Data	
2	Girl	16 to 20	College	Government	No	Yes	Low	Mid	Wifi	
3	Girl	11 to 15	School	Non Government	No	Yes	Low	Mid	Mobile Data	
4	Girl	16 to 20	School	Non Government	No	Yes	Low	Poor	Mobile Data	
...	...	...	...	...	...	...	...	...	...	
1200	Girl	16 to 20	College	Non Government	No	Yes	Low	Mid	Wifi	
1201	Girl	16 to 20	College	Non Government	No	No	High	Mid	Wifi	
1202	Boy	11 to 15	School	Non Government	No	Yes	Low	Mid	Mobile Data	
1203	Girl	16 to 20	College	Non Government	No	No	Low	Mid	Wifi	
1204	Girl	11 to 15	School	Non Government	No	Yes	Low	Poor	Mobile Data	

1205 rows × 14 columns

1. Draw an appropriate graph to count the number of boy and girl students who participated in the survey.

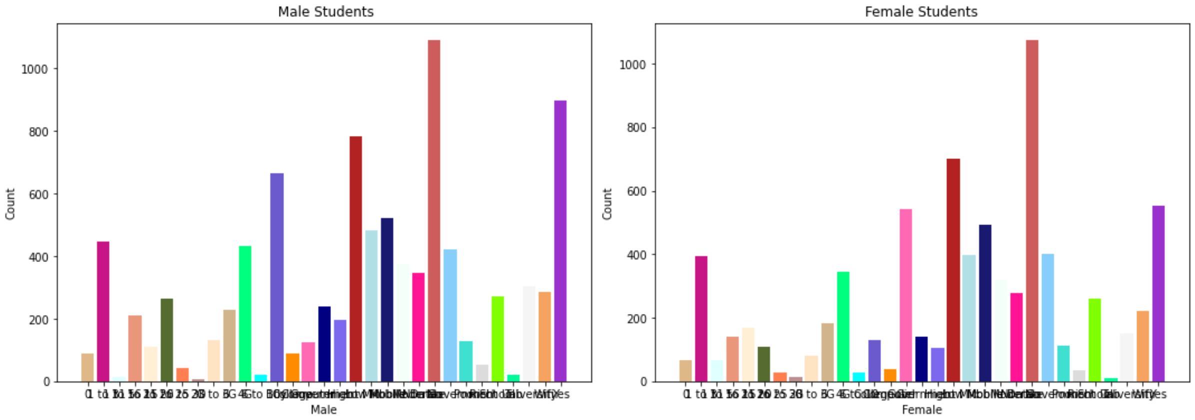
In [17]:

```
df.Gender.value_counts()
```

Out[17]:

```
Boy     663
Girl    542
Name: Gender, dtype: int64
```

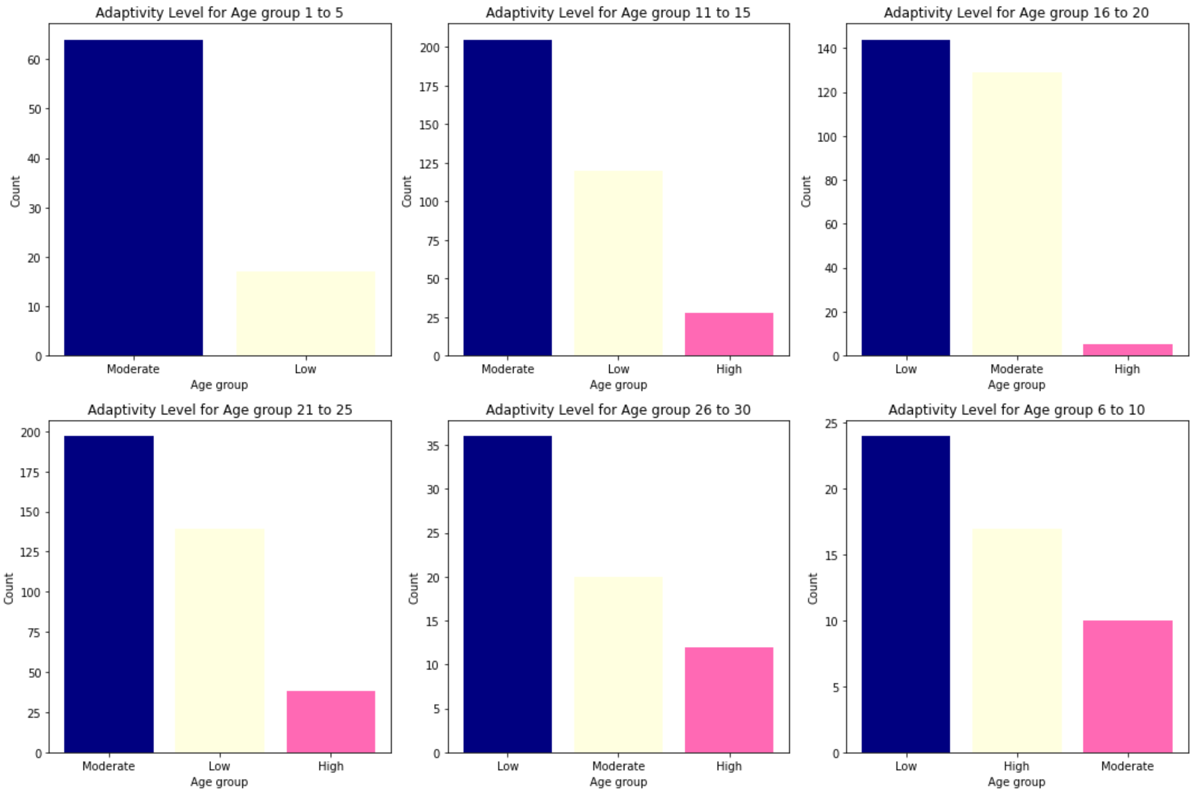
In [34]:
```
random.shuffle(colors)
plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
un, count = np.unique(data[data.Gender == "Boy"].values,return_counts=True)
plt.bar(un, count, color=colors)
plt.title(" Male Students")
plt.xlabel("Male")
plt.ylabel("Count")

plt.subplot(2,2,2)
un, count = np.unique(data[data.Gender == "Girl"].values,return_counts=True)
plt.bar(un, count, color=colors)
plt.title("Female Students")
plt.xlabel("Female")
plt.ylabel("Count")

plt.tight_layout()
```



1. Draw an appropriate graph to count the number of students in different age groups.

In [20]:

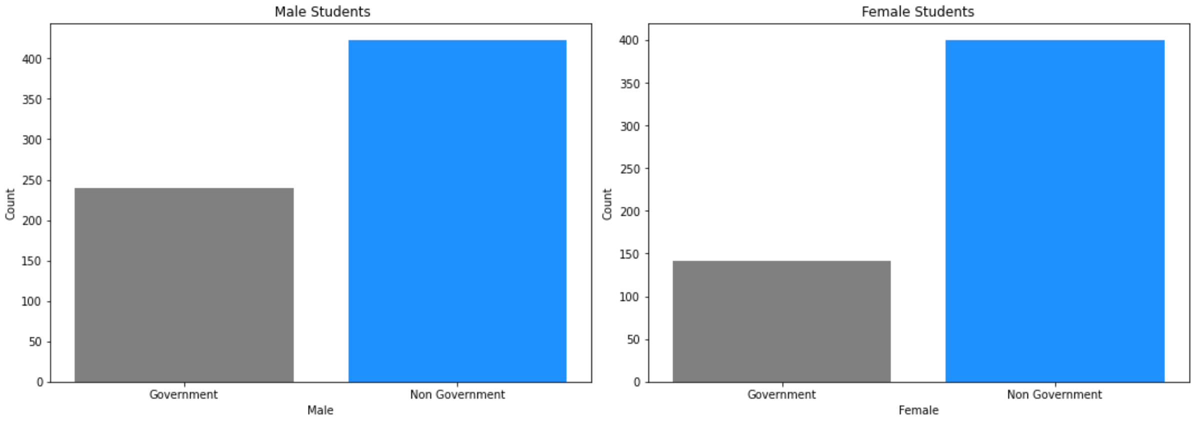
```
random.shuffle(colors)
plt.figure(figsize=(15,10))
graphs = 1
for i in np.unique(data.Age.values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data.Age == i]["Adaptivity Level"].values, retu
rn_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("students in different Age group " + i)
    plt.xlabel("Age group")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



1. Draw an appropriate graph to count the number of boy and girl students separately studying in different education institutions.

In [35]:

```
random.shuffle(colors)
plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
un, count = np.unique(data[data.Gender == "Boy"]["Institution Type"].values,return_counts=True)
plt.bar(un, count, color=colors)
plt.title(" Male Students")
plt.xlabel("Male")
plt.ylabel("Count")

plt.subplot(2,2,2)
un, count = np.unique(data[data.Gender == "Girl"]["Institution Type"].values,return_counts=True)
plt.bar(un, count, color=colors)
plt.title("Female Students")
plt.xlabel("Female")
plt.ylabel("Count")

plt.tight_layout()
```



1. On a scale of 100%, draw an appropriate graph to project the Adaptivity Level of students.

In [19]:

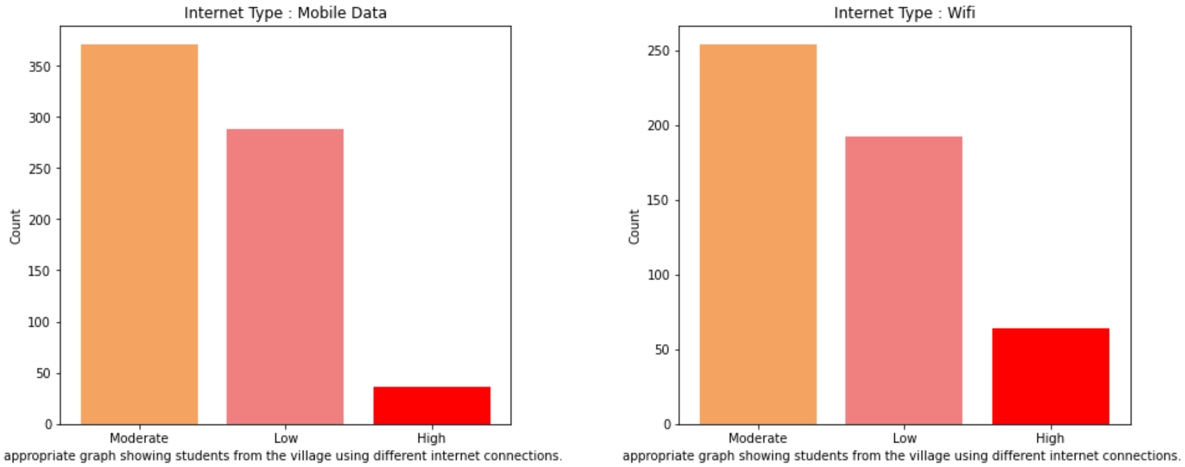
1. Use an appropriate graph to compare the performance of the students from town and from village.

In [ ]:

1. Draw an appropriate graph showing students from the village using different internet connections.

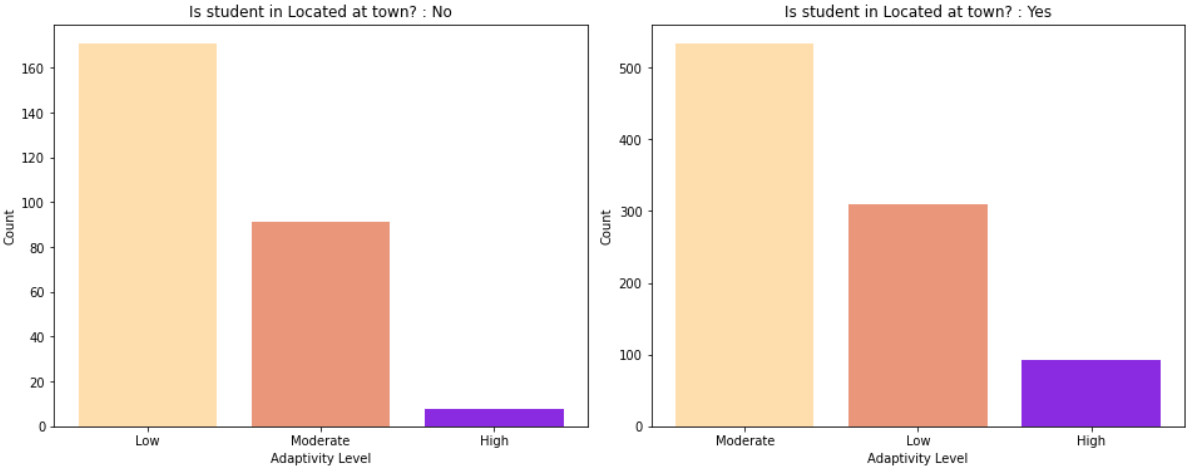
In [29]:

```
random.shuffle(colors)
plt.figure(figsize=(20,10))
graphs = 1
for i in np.unique(data["Internet Type"].values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data["Internet Type"] == i]["Adaptivity Level"].values, return_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("Internet Type : " + i)
    plt.xlabel("appropriate graph showing students from the village using different internet connections. ")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



1. Compare the adaptivity levels of students from town and students from villages.

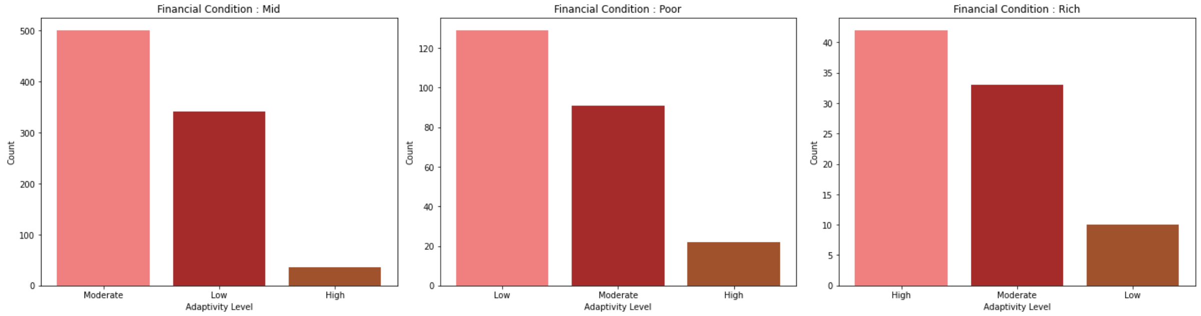
In [26]:
```
random.shuffle(colors)
plt.figure(figsize=(20,10))
graphs = 1
for i in np.unique(data["Location"].values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data["Location"] == i]["Adaptivity Level"].valu
es, return_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("Is student in Located at town? : " + i)
    plt.xlabel("Adaptivity Level")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



1. Who has a better Adaptivity Level under different financial conditions?

In [27]:

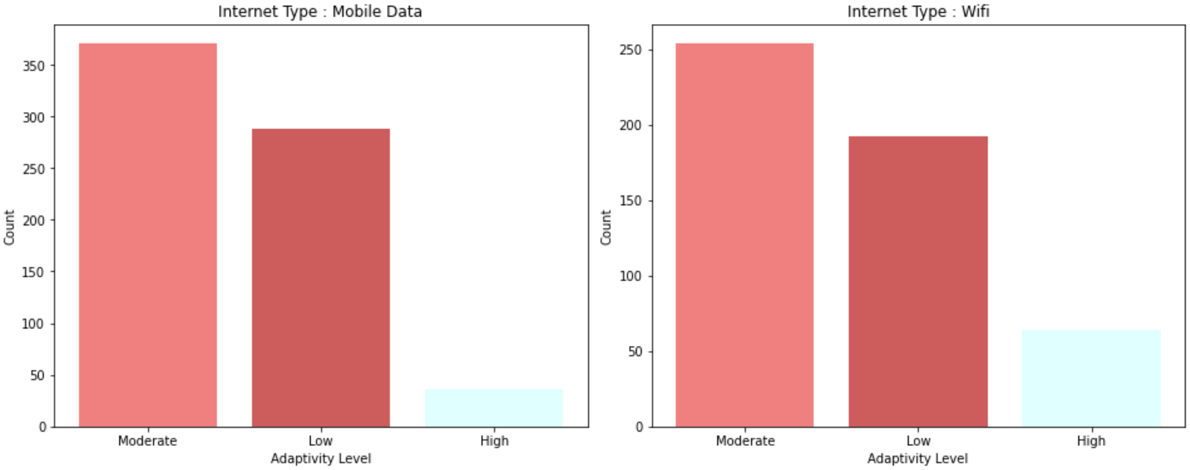
```python
random.shuffle(colors)
plt.figure(figsize=(20,10))
graphs = 1
for i in np.unique(data["Financial Condition"].values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data["Financial Condition"] == i]["Adaptivity Level"].values, return_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("Financial Condition : " + i)
    plt.xlabel("Adaptivity Level")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



1. Does usage of mobile data vs wifi affect adaptivity levels of students?

In [37]:

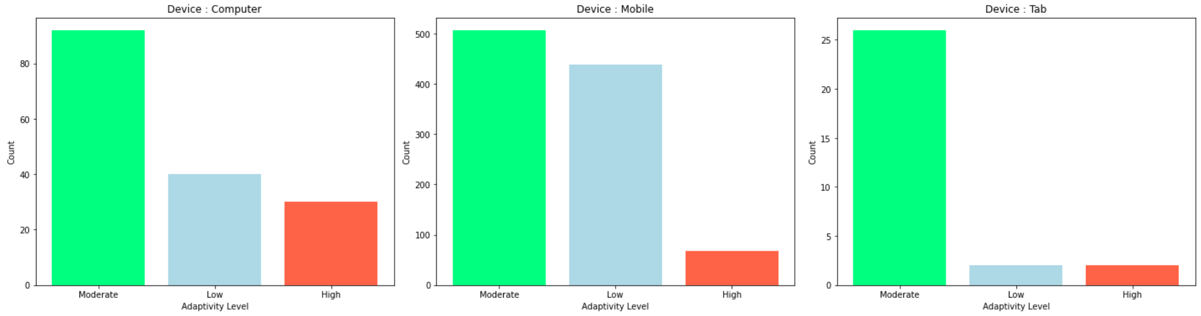
```
random.shuffle(colors)
plt.figure(figsize=(20,10))
graphs = 1
for i in np.unique(data["Internet Type"].values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data["Internet Type"] == i]["Adaptivity Level"]
.values, return_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("Internet Type : " + i)
    plt.xlabel("Adaptivity Level")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



1. Compare Financial Condition with Device owned. Is it true that poor people cannot afford a laptop?

In [36]:

```
random.shuffle(colors)
plt.figure(figsize=(20,10))
graphs = 1
for i in np.unique(data["Device"].values):
    plt.subplot(2,3,graphs)
    un, count = np.unique(data[data["Device"] == i]["Adaptivity Level"].values
, return_counts=True)
    count, un = zip(*sorted(zip(count, un), reverse=True))
    plt.bar(un, count, color = colors)
    plt.title("Device : " + i)
    plt.xlabel("Adaptivity Level")
    plt.ylabel("Count")
    graphs = graphs + 1
plt.tight_layout()
```



In [ ]:

```
!jupyter nbconvert --to html /content/19BCE7346_TARAN_NLP_Assignment_4.ipyn
b
```